Assignments

Below you find a list of possible assignments related to the topics discussed in the lecture. Please hand in (signed and encrypted by mail to a.t.huelsing@tue.nl with the subject starting with "[AppliedCrypto]"; see, e.g., http://www.win.tue.nl/applied_crypto/2018/ for the public key) your first, second, and third preferred choice (indicated by topic number) before **December 19th (Wednesday) 23:59**. Whenever possible you will get one of your choices. The topic will be assigned to you before Christmas.

Each assignment asks you to study some literature on a cryptographic scheme, system,or attack, and to write a report of **6 pages**; using the Springer style file from http://www.springer.com/computer/lncs?SGWID=0-164-6-793341-0. These assignments are individual.

If not explicitly stated otherwise, your report should focus on the cryptographic-systems aspects of the paper. This means that your report should cover all relevant cryptographic aspects of the entire system. You should start with a description of what functionality the system provides and what it means in this context for a system to be secure. Then you should explain how the system implements the functionality (what cryptographic primitives are used where, how do they interact etc). You should provide argumentation and motivation why the system is secure or not.

I am not interested in a repetition of the core claims of the paper. What I want to see is that you try to capture the intuition behind the system and its security (or behind the attack). If the lecture already dealt with your assigned topic, your work should clearly go beyond what was explained in the lecture. Just summarizing lecture content is not enough to pass.

After this summary of the paper add your own opinion in a discussion part. Answer questions like

- Is this system well designed? Does it solve the problems it aimed to solve?
- Do you see possible (cryptographic) attacks?
- What is the practical relevance of the attack and why? (If you write about an attack)
- Do you see countermeasures?
- What could be improved?

In addition to this answer the specific questions provided for your topic.

A note on copying, plagiarism and citing.

Copying text not written by yourself is not allowed (with the exception of *citing*, see below). This holds for text from the material to be studied, and also from other (web-)publications. This rule includes *plagiarism* (copying text written by somebody else while making it appear as if it is written by you), but is not limited to that, i.e., it is not allowed to copy substantial amounts of text from other sources, even when you do add proper referencing. You should write yourself.

Citing in the sense of literally citing is a different matter. While it is of course allowed, it should be used with caution, if used at all. Literally citing can be defined as: copying a small amount of text with proper referencing, that supports, and not replaces, your own text.

Citing in the sense of providing references underpinning your statements is not only allowed but mandatory. Providing proper references to all your sources is mandatory, too.

The report will be judged on how well you cover the topic, sensible argumentation, and also on readability. While we are not judging your English, we cannot give credit for what we do not understand.

Deadline: **February 1st (Friday), 2019 at 23:59**.

Allowed languages: English.

Deliver your paper in electronic form by e-mail, signed and encrypted, to a.t.huelsing@tue.nl as a pdf document named "[LAST_NAME][FIRST_NAME]Topic[TOPIC_NUMBER].pdf", again the subject line starting with "[AppliedCrypto]". Make sure that your public key is available, either in a public directory or attached to your email.

You will receive feedback and your grade by e-mail. The grade for the assignment contributes 50% of your final grade.

Most cited papers are available online (sometimes only from within the TU/e network). Books should be in the TU/e library. If you can't find papers and/or books, write me an email.

Coding assignments
This year I decided to alternatively offer the option to do a coding assignment. A coding assignment consists of demonstrating an attack against a cryptographic protocol that we covered in the lecture or that is closely related to the topics of the lecture. Possible attacks are:

- All the attacks against TLS, including Bleichenbacher attack.
- Invalid point and twist attacks against ECC.
- Reaction attacks against lattices/codes/isogenies.
- Any of the attacks in the writing assignments below.

In a first step, you are asked to implement a server that provides an interface implementing a vulnerable, possibly simplified version of the target cryptographic protocol. This has to be accompanied with software that demonstrates the attack. In a second step you are asked to advance server and attack software to a more realistic setting. For example in the context of TLS attacks this could be implementing a vulnerable TLS server and implementing attack software that runs in the context of a victim.

Your code has to be well documented and must be accompanied by a short report explaining the attack you implemented and its general working.

The result will be judged based on functionality, how realistic the final demonstrator and server are, and on the content of the report. Of course we will take into account the complexity of the implemented attack.

Of course the same rules on copying, citing, and plagiarism as above apply to the accompanying documents and the code.

While we do not have explicit limitations and the allowed programming languages, we can only provide support in Python, Java, and C/C++.

If you want to do a coding assignment, please contact me **before** the deadline and make an appointment to discuss the details.

Individual Literature Assignments

The following list contains proposals from my side. If you have a topic that you are interested in but for which the list contains no assignment, you can contact me **before** the deadline. Please name a paper and a question related to that work that you want to answer. I will decide if this topic meets my requirements. The topic has to be closely related to one of the topics of the lecture.

Topics

1. Replacing PKI. Study "Certificate-based Encryption" by Gentry (http://eprint.iacr.org/2003/183.pdf). Describe the cryptographic tools used and compare it to traditional PKI. Would you replace PKI with this? Why?
2. Replacing PKI. Study "Certicateless Public Key Cryptography" by Al-Riyami and Paterson (http://eprint.iacr.org/2003/126.pdf). Describe the cryptographic tools used and compare it traditional PKI. Would you replace PKI with this? Why?
3. E-mail security: EFAIL. Study the EFAIL attack (start at https://efail.de/). Discuss the two different attack variants. Research and discuss countermeasures that prevent the attack.
4. Password-hashing. There recently was a password hashing competition running (https://password-hashing.net/) that tried to find a new password hashing scheme to replace old schemes like bcrypt or PBKDF2. Describe the security requirements a password-hashing scheme should fulfill. Sketch the winner Argon2, describe how / why this algorithm met the requirements, and what issues where found recently.
5. eCash: Double-spending in bitcoin. Get yourself an overview of the bitcoin protocol. Then look into Double-Spending-Attacks ("Two Bitcoins at the Price of One? Double-Spending Attacks on Fast Payments in Bitcoin" by Karame, Androulaki, and Capkun). Describe the problem, the attack, and the proposed counter-measure. Then search for other counter-measures in literature and describe at least two.
6. eCash: Proof-Of-Stake. Read https://eprint.iacr.org/2016/889.pdf. Explain the concept of Proof-Of-Stake protocols. What is the difference to traditional bitcoin? How is security reasoned?
7. Quantum algorithms. Read Grover's original paper on his algorithm (http://arxiv.org/abs/quant-ph/9605043). Explain the algorithm such that it is accessible by a master student in computer science. Also, discuss the implications for cryptography and the lower bound results.
8. Privacy-preserving bitcoin. Study zero-cash (http://zerocash-project.org/), read their scientific paper and compare it to bitcoin. What is improved, what is getting worse? Describe the protocol, discuss the claimed security properties and if they are achieved.
9. Key transport. Take a look at A. Dent: "A designer's guide to KEMs" (https://eprint.iacr.org/2002/174/20051031:161821). Explain the concept of the KEM/DEM terminology, summarize the generic transformations presented, and explain one of the security reductions in detail. For the latter, focus on intuition and main ideas rather than technical details.
10. Mass surveillance. Explain the concept of Big-key cryptography. Which problem is tackled and how? Possible starting points are P. Rogaway: "The moral character of cryptographic work" (http://web.cs.ucdavis.edu/~rogaway/papers/moral.pdf) and Bellare, Kane, Rogaway: "Big-Key Symmetric Encryption: Resisting Key Exfiltration" (https://eprint.iacr.org/2016/541.pdf).

11. Tor. The Tor Network does not use, e.g., steganography in order to hide Tor traffic; however, there are mechanisms to prevent governments from blocking Tor. Study the security of Tor in respect to censorship avoidance (e.g., blocking of Tor in China). What countermeasures are discussed in literature (e.g., "How China Is Blocking Tor" by Winter and Lindskog)?
12. Tor: Elligator. Read https://elligator.cr.yp.to/elligator-20130828.pdf. How does the scheme work. What does it achieve? How is this related to Tor (give context). What are possible limitations / attacks.
13. OTR. Study the TextSecure OTR protocol; describe the changes from version 1 to version 2 and the relation to the Axololt protocol. Use https://github.com/WhisperSystems/TextSecure/wiki as starting point when searching for literature.
14. Private messaging. Read "A Formal Security Analysis of the Signal Messaging Protocol" (https://eprint.iacr.org/2016/1013). Explain the security model used and give an intuition for the main proofs.
15. TLS. Study TLS 1.3. Give a summary of the functionality (modes) provided. Give a clear description of the differences compared to previous versions of the protocol, and provide reasons for these differences. What attacks are prevented, what improvements are made? SSLv3 is described in RFC 6101, TLS versions are in RFC 2246, 4346, 5246, and 8446; the last one being TLSv1.3. Also have a look at Eric Rescorla: "SSL and TLS, Designing and Building Secure Systems".
16. TLS attacks. Study the weaknesses of TLS as pointed out in the paper "Lucky Thirteen: Breaking the TLS and DTLS Record Protocols" by AlFardan and Paterson, 2013. Include a description of their attack and suggested countermeasures.
17. TLS attacks. Study the "Logjam" attack on TLS as described in the paper "Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice" by Adrian et. al., 2015. Focus on the protocol-related aspect of the attack and include a description of suggested countermeasures.
18. TLS security. Summarize the paper "Indiscreet Logs: Persistent Diffie-Hellman Backdoors in TLS" (https://eprint.iacr.org/2016/999.pdf). Specifically explain how the the trapdoor with composite moduli works and how the backdoor was found in practice.
19. TLS security: Reaction attacks. Read the original paper on reaction attacks (https://www.schneier.com/academic/archives/1999/06/reaction_attacks_aga.html). Explain the attack. Explain how and under which conditions the attack works.
20. Attacks on RSA. Study the recent ROBOT attack (https://robotattack.org/). Read the scientific paper by the authors, explain the attack and its impact. Discuss countermeasures.
21. Authenticated encryption: Attacks against OCB2. Study the recent attacks against OCB2 published in https://eprint.iacr.org/2018/1087, https://eprint.iacr.org/2018/1090, and https://eprint.iacr.org/2018/1040. Explain how the attacks work and what their impact is. Discuss countermeasures.
22. Passwords: Honeywords. Describe the Honeywords method used to improve the security of hashed password files from (https://people.csail.mit.edu/rivest/pubs/JR13.pdf). Discuss its strengths and weaknesses and how this system makes hashed password files more secure against attackers.
23. Lattice-based Crypto: LWE. Describe the LWE encryption scheme and the underlying security assumption (See the original paper http://www.cims.nyu.edu/~regev/papers/qcrypto.pdf and the survey http://www.cims.nyu.edu/~regev/papers/lwesurvey.pdf). What does this have to do with lattices? In addition describe how Dual LWE Encryption works.

24. Lattice-based Crypto: GGH. Describe how the GGH signature scheme works and how to attack the scheme by using "Learning a Parallelepiped: Cyptanalysis of GGH and NTRU Signatures" (http://www.cims.nyu.edu/~regev/papers/gghattack.pdf). For more information on the GGH signature scheme, see http://groups.csail.mit.edu/cis/pubs/shafi/1997-lncs-ggh.pdf.
25. Lattice-based Crypto: NTRU. Describe how the NTRU cryptosystem works (See for instance the PKCS Tutorial on https://www.securityinnovation.com/products/encryption-libraries/ntru-crypto/ntru-resources.html and the original paper on https://www.securityinnovation.com/uploads/Crypto/ANTS97.pdf). Describe how a Meet-in-the-Middle attack on the NTRU secret key pair works (https://www.securityinnovation.com/uploads/Crypto/NTRUTech004v2.pdf).
26. Hash-based signatures: PRUNE-HORST. Read the NIST-submission document for PRUNE-HORST (https://github.com/gravity-postquantum/prune-horst/blob/master/Supporting_Documentation/submission.pdf). Describe the scheme, the security reasoning and explain how it differs from the original HORST scheme.